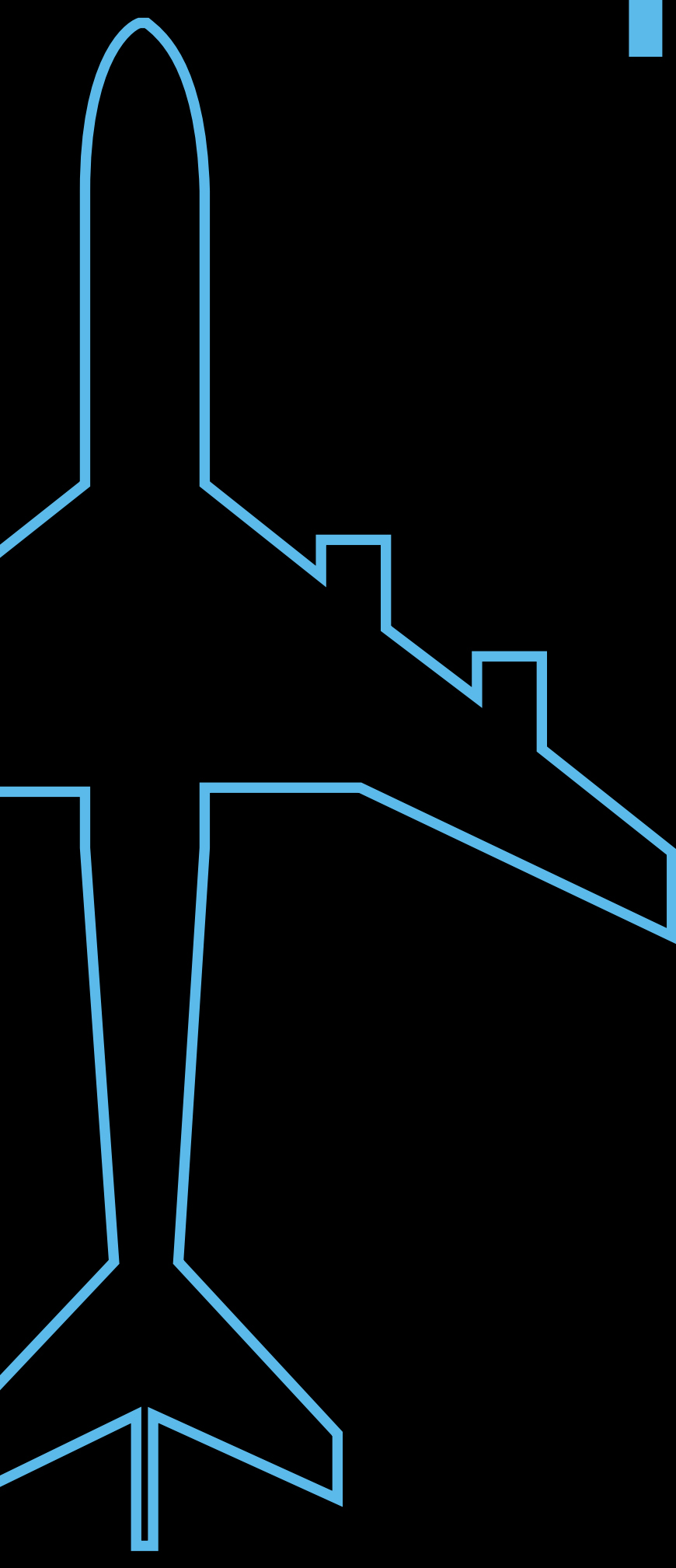


# **SERVERLESS IN THE JAVA ECOSYSTEM**

**Pratik Patel @prpatel**

**@prpatel**



**PRATIK PATEL — @PRRPATEL**

**CTO TRIPLINGO**

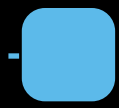
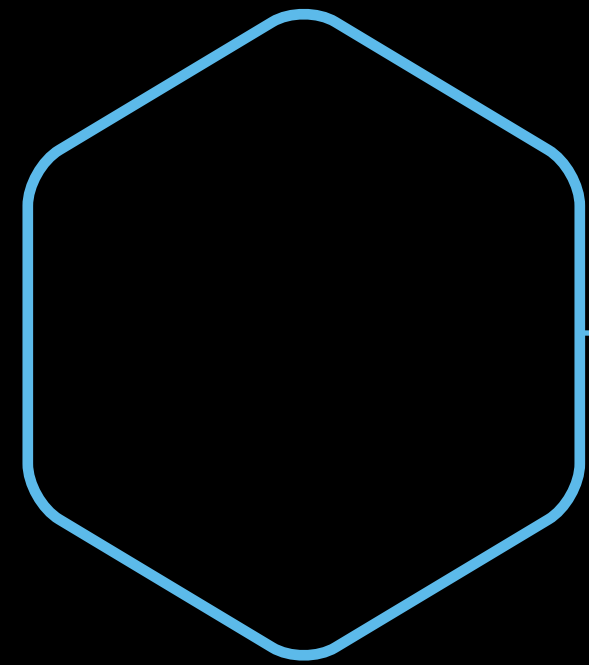
**Java Champion**

**JavaScript Troublemaker**

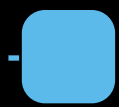
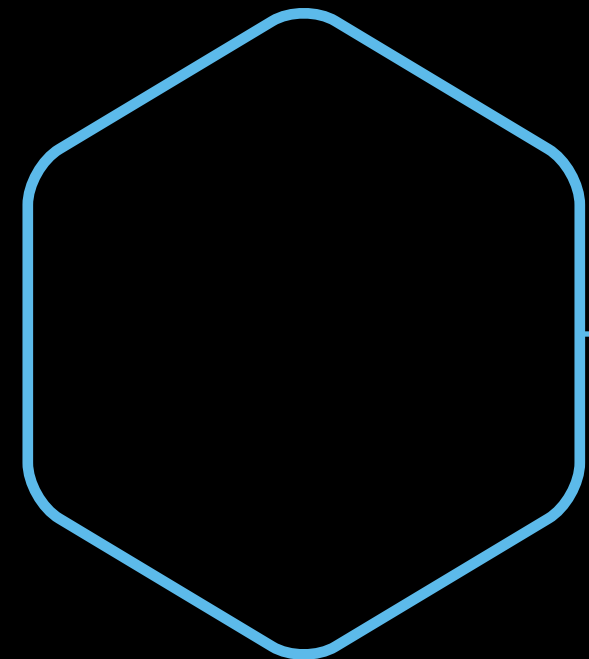
**PYTHON Hacker**

**Founder, PERL recovery group**

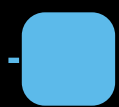
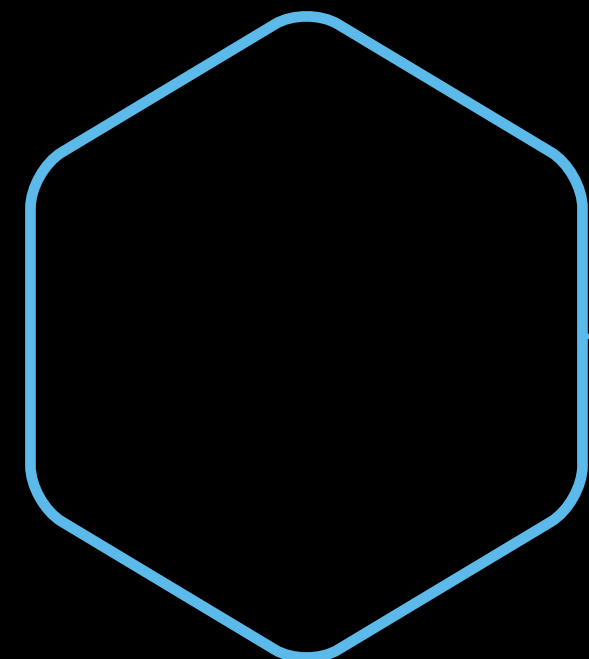
**@prpatel**



**WHAT IS SERVERLESS?**



**ARCHITECTURE**



**ECOSYSTEM**

# SERVERLESS



**A VERY BAD NAME**



# IN A NUTSHELL



**Kelsey Hightower** 

@kelseyhightower



We asked 10 people to define serverless. We got 10,000 different answers.

12:17 PM · Jan 31, 2018

# RESPONSES WERE EVEN BETTER!



**Emad Benjamin** @vmjavabook · Jan 31

Replying to [@kelseyhightower](#)

We asked developers to write one line of code, instead we got 10k containers



# RESPONSES WERE EVEN BETTER!



**Greg**   
@waltisfrozen

Replying to [@kelseyhightower](#)

Serverless is Geocities for the Enterprise.

# NOT GEOCITIES

5-7 Feb 2018

[Register](#) [Speakers](#) [Talks](#) [Schedule](#) [Training](#) [VM Summit](#) [Video](#) [Kidz](#)

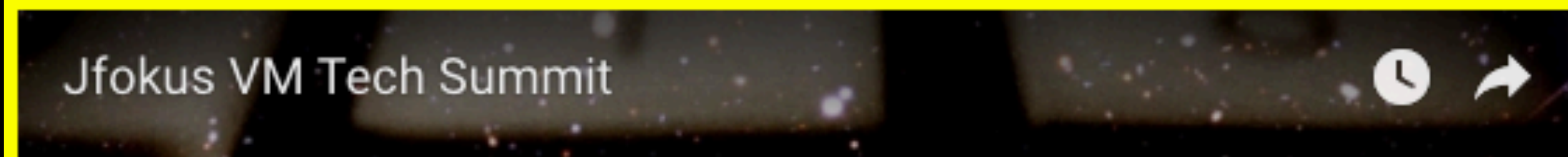
[Register](#) [Speakers](#) [Talks](#) [Schedule](#) [Training](#) [VM Summit](#) [Video](#) [Kidz](#)

Jfokus Developers Conference 5-7 February 2018

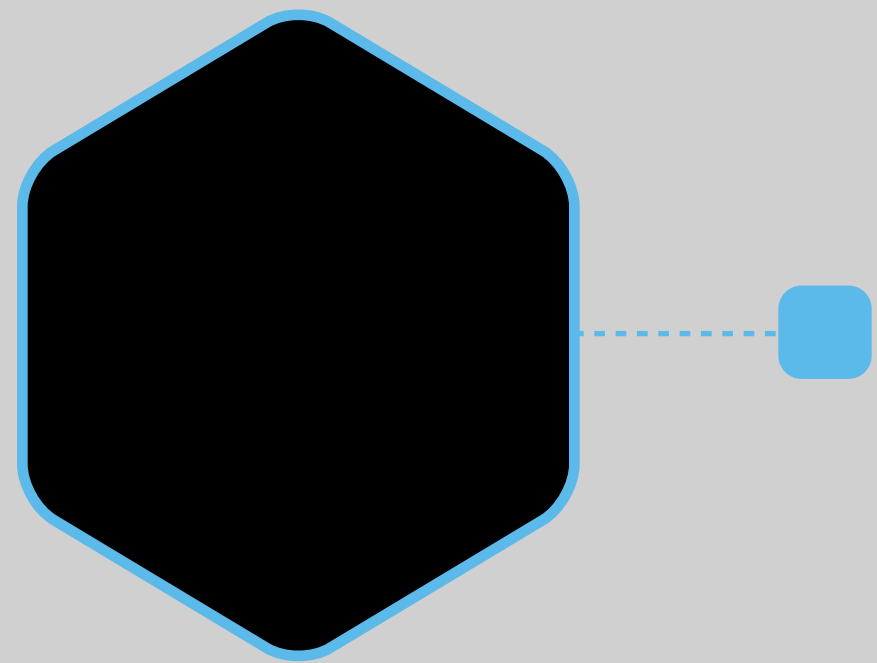
Jfokus is all about developers! Java SE & Java EE, Frontend & Web. Android & Mobile, Continuous Delivery & DevOps, Internet of Things, Cloud & Big Data, Future & Trends, Alt.JVM Languages like Scala, Clojure & many more, Agile development. And super heroes...



[Monday 9.00 LIVE Stream - Jfokus VM Tech Summit](#)

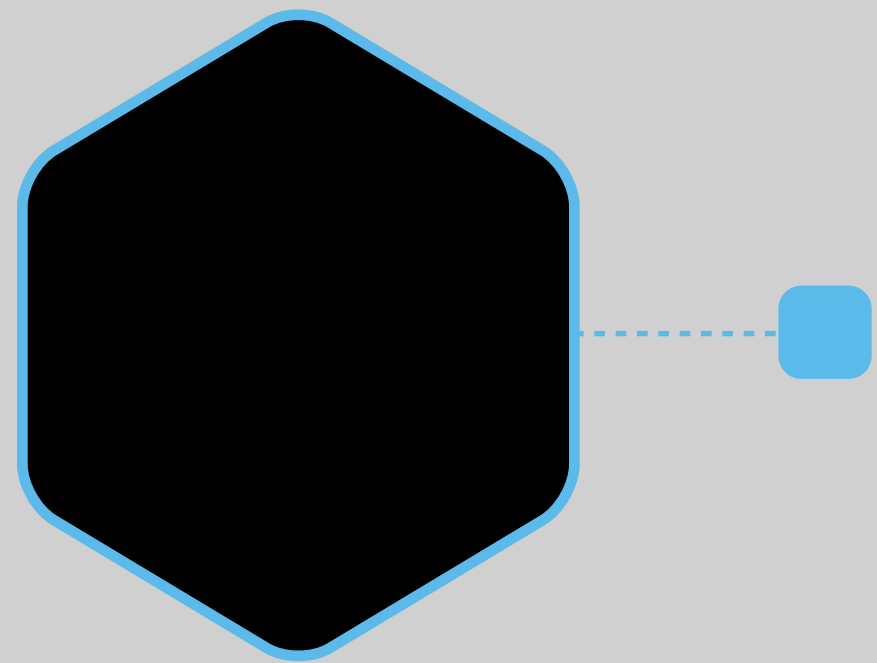


@prpatel



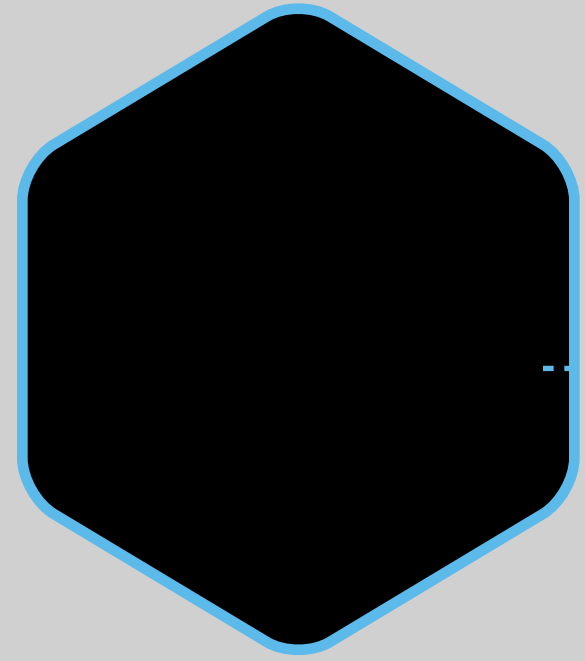
**Yes, There Are Still  
Servers**

**SERVERLESS**

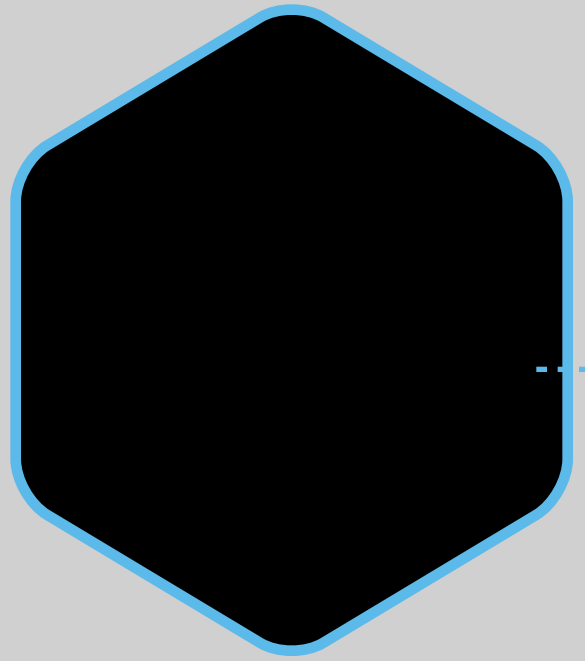


**New way to pay for and  
work with servers**

**SERVERLESS**



**Cheaper**



**Easier**

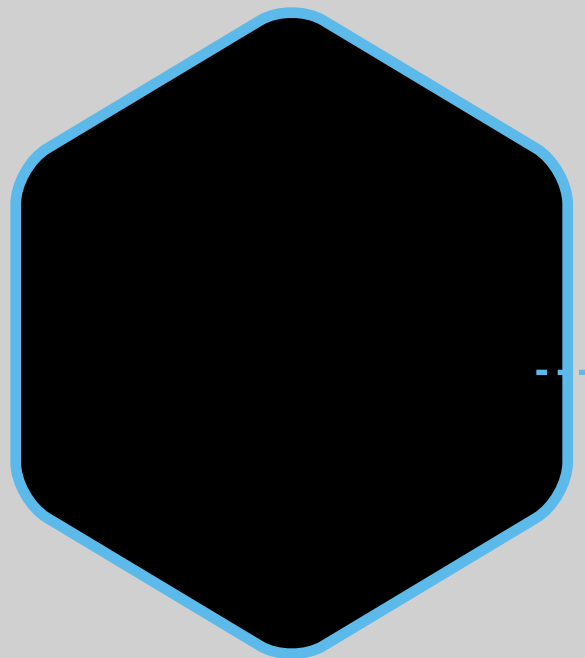
**WHAT'S THE POINT?**



# Aliases AKA



**Functions as a Service**



**Cloud Functions**

**AKA**

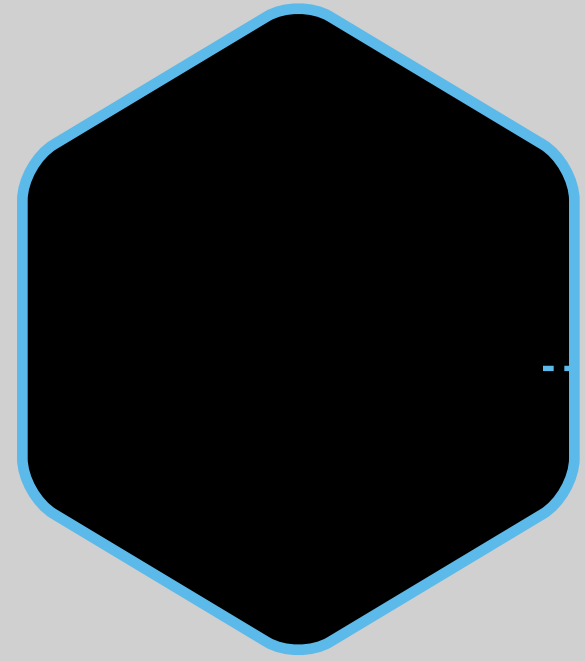


## DEFINITION

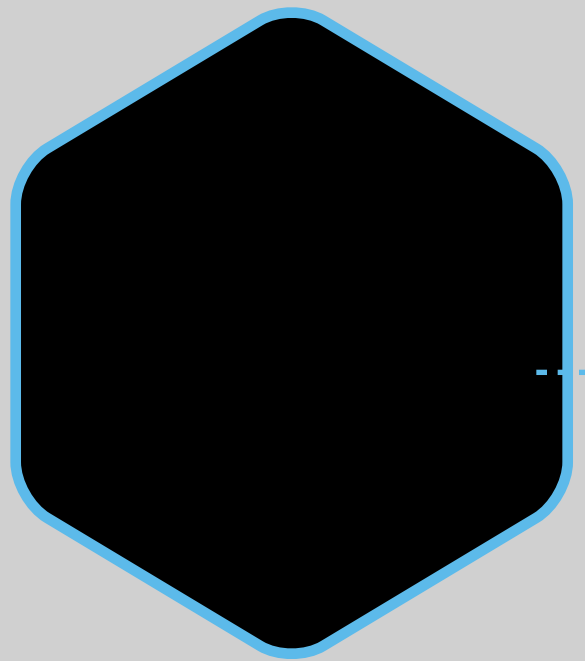
**Run code without  
provisioning or managing  
servers**



**All the DEV  
no  
DevOps**



**Scalability**



**Availability**

**SERVERLESS TAKES CARE OF IT**



**But is it also a (new)  
architectural model?**

# CONSUMER COMPUTING

MICROCOMPUTER → SAAS

# ENTERPRISE COMPUTING?

IAAS



SERVERLESS

**APPS**

**CONTAINERS**

**PLATFORM**

**INFRASTRUCTURE**

**APPS**

**FAAS**

**CONTAINERS**

**PLATFORM**

**INFRASTRUCTURE**

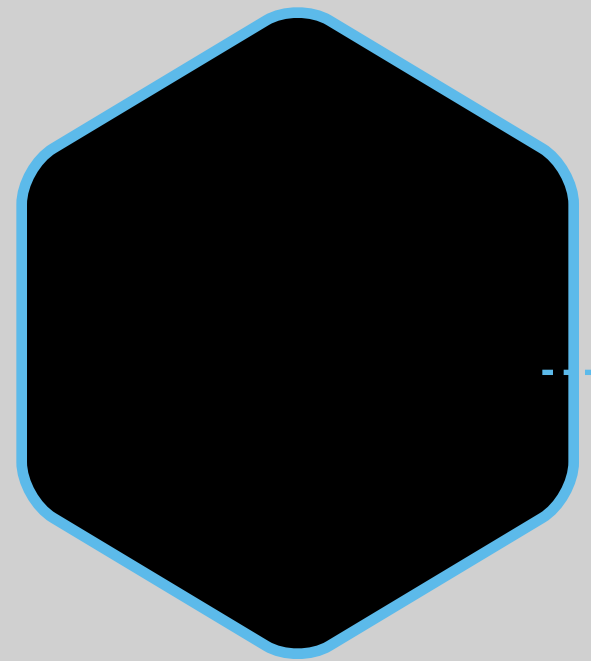
**APPS**

**FAAS**

**INFRASTRUCTURE –  
SOMEONE ELSE'S JOB**

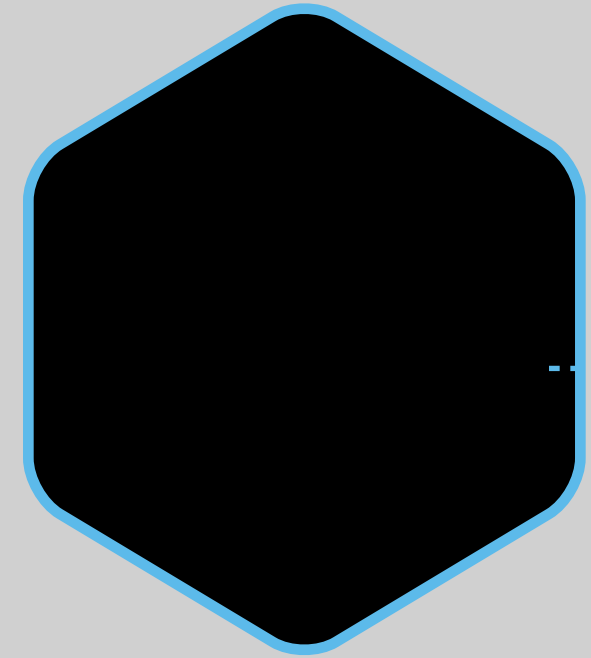


**Focus on building apps**

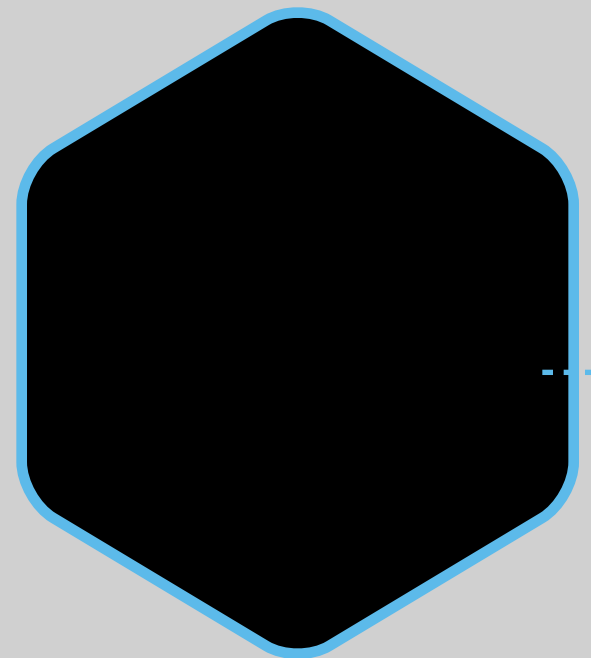


**Docker? What's that?**

**NEW ARCHITECTURE**



**Use existing FaaS's**

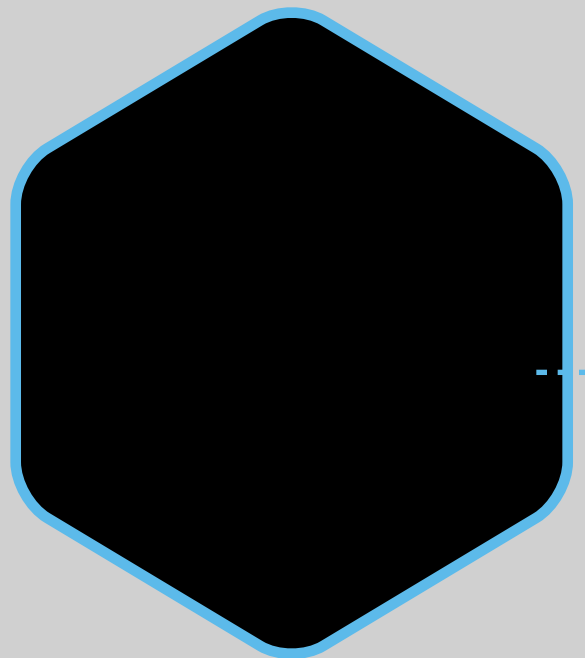


**Write own FaaS**

**NEW ARCHITECTURE**



**Event-Driven**



**Events can be almost  
anything**

**NEW ARCHITECTURE**

# FAAS EVENTS

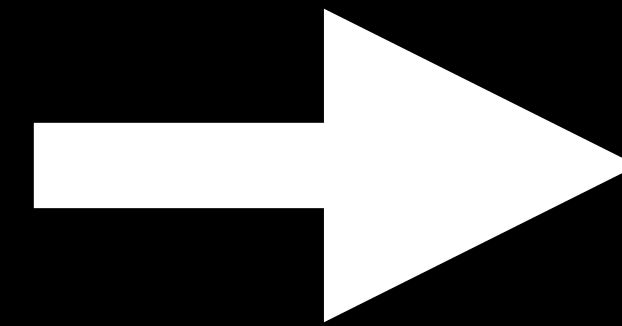
Message (queue)

HTTP call

FS or DB monitor

Inbound Email

Alexa



INVOKE  
FUNCTION

**THE REVOLUTION  
WILL NOT BE  
TELEVISED**

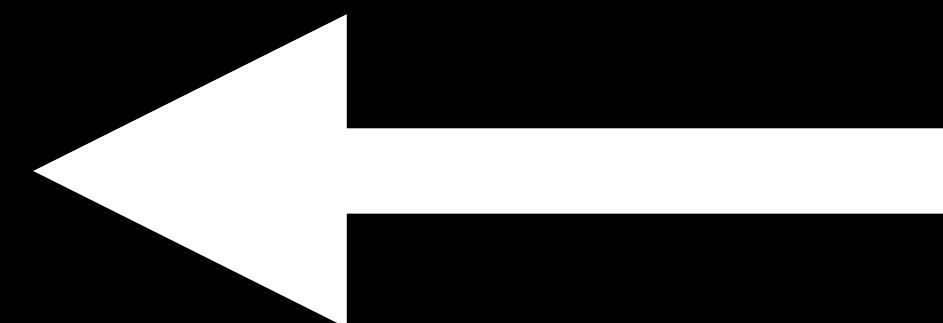
# WE (YOU?) ALREADY DOING THIS



Google FaaS

*Image OCR*

*Translate*



*Our own FaaS*

# WE (YOU?) ALREADY DOING THIS

my  
app



- *Twilio for SMS*
- *Stripe (payments)*
- *Image resize*
- *Send/format email*
- *Auth with Okta*
- *... lots more!*

# ARCHITECTURAL SHIFT

Include lib in  
project



Call a FaaS

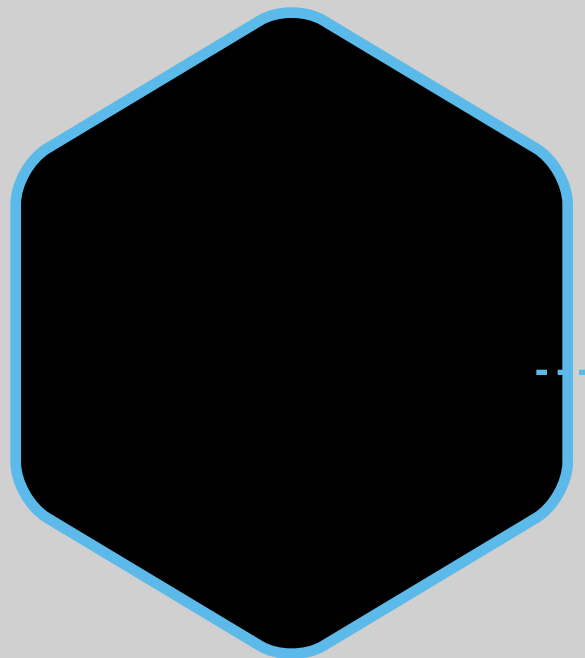
Include lib in  
project



Call a FaaS



**Small (a unit of work)**

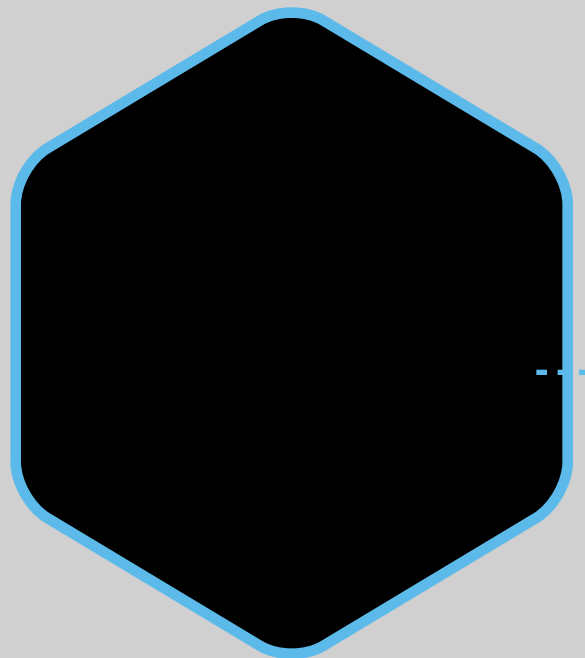


**Single-purpose**

**FAAS**



**Short Running**



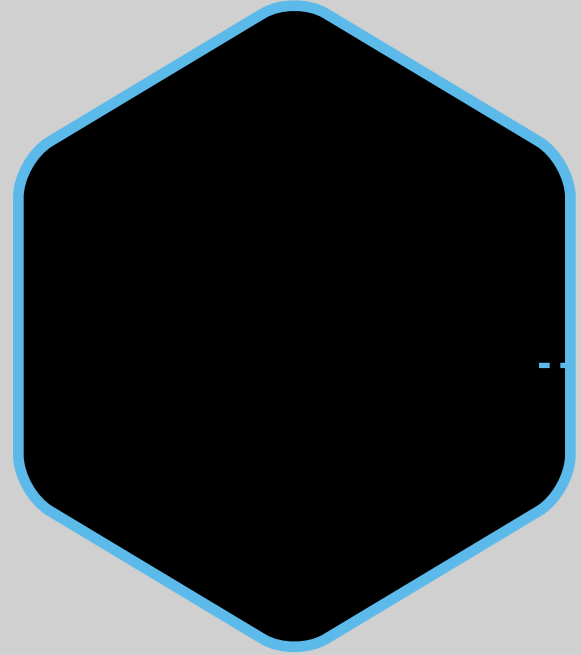
**Stateless**

**FAAS**

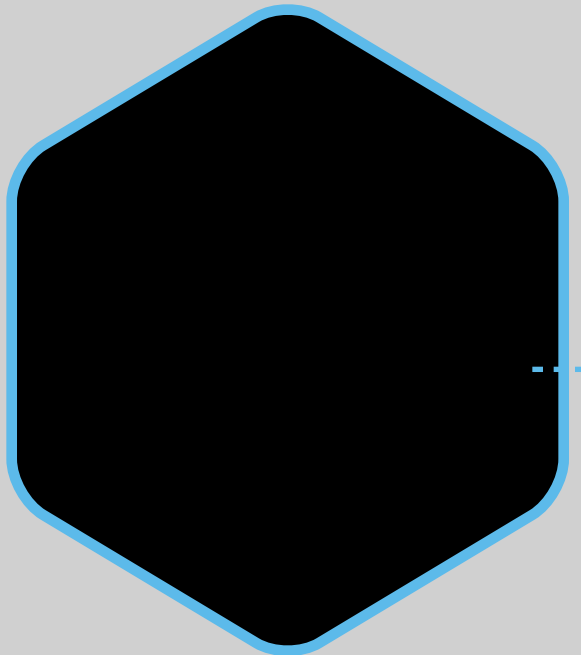
# FaaS Implementations - Java

# INVOKATION

Can be invoked by any platform (doesn't have to be Java)



**Fn Project**

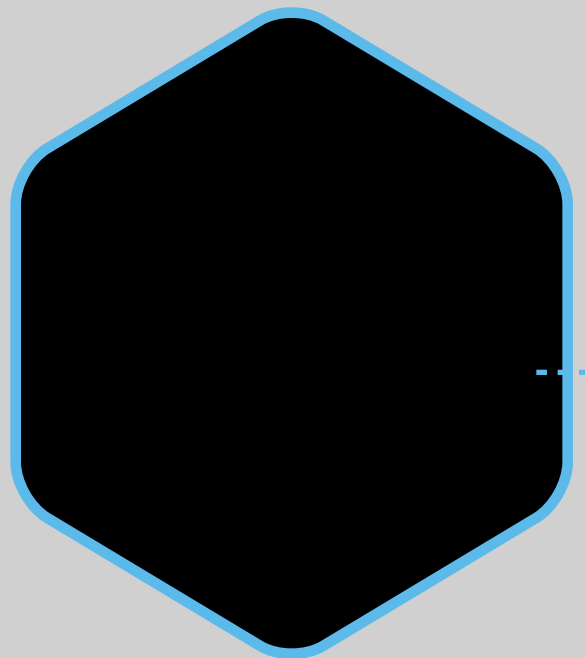


**<https://github.com/fnproject/fn>**

**FAAS IMPLEMENTATIONS**

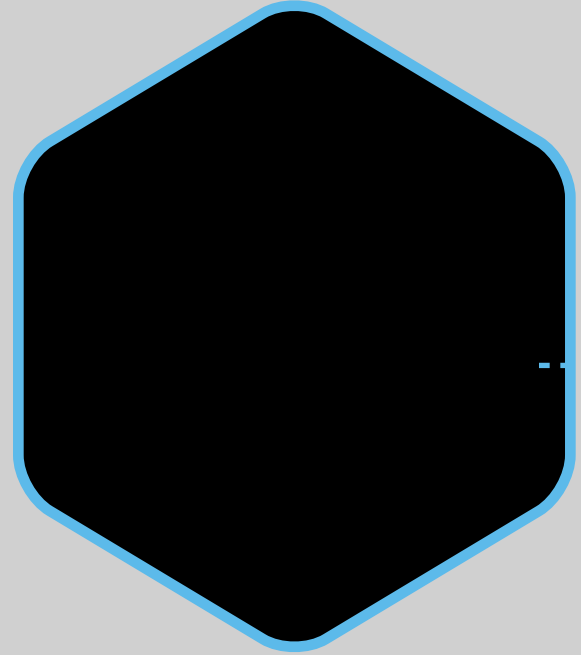


**OpenWhisk**

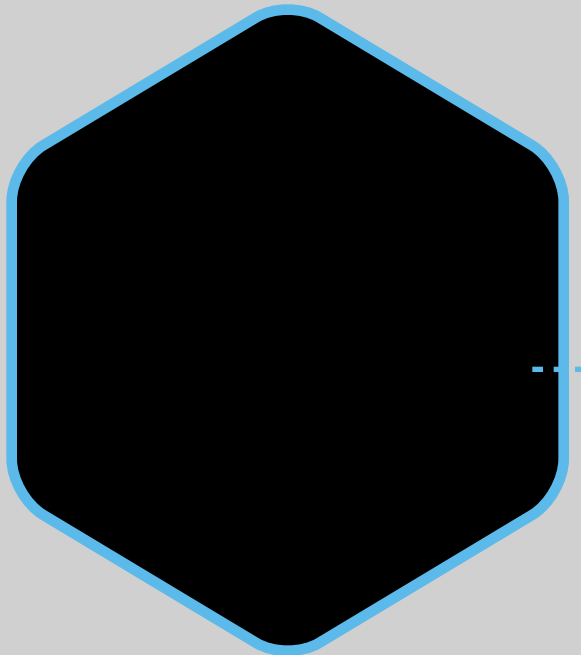


**[http://  
openwhisk.incubator.apache.org/](http://openwhisk.incubator.apache.org/)**

**FAAS IMPLEMENTATIONS**



**Amazon Lambda**



**<https://aws.amazon.com/lambda/>**

**FAAS IMPLEMENTATIONS**



**DEMO**

**Fn Project**


Two stylized blue gears are positioned on the left side of the slide. The larger gear is in the foreground, and a smaller one is partially visible behind it to the right.

(make sure docker is running)

```
fn start
```

```
fn init --runtime java hello
```

```
fn run  
// Hello World
```



```
├─ func.yaml
├─ pom.xml
└─ src
    ├─ main
    │   └─ java
    │       └─ com
    │           └─ example
    │               └─ fn
    │                   └─ HelloFunction.java
    └─ test
        └─ java
            └─ com
                └─ example
                    └─ fn
                        └─ HelloFunctionTest.java
```



```
package com.example.fn;

public class HelloFunction {

    public String handleRequest(String input) {
        String name = (input == null || input.isEmpty()) ? "world" : input;

        return "Hello, " + name + "!";
    }
}
```



**fn run**  
**- just a local**  
**execution**

A decorative graphic on the left side of the slide. It features a large, light blue gear icon at the top, with several blue lines of varying thicknesses and shapes extending downwards and outwards, resembling a stylized circuit or network diagram.

```
export FN_REGISTRY=prpate1
```

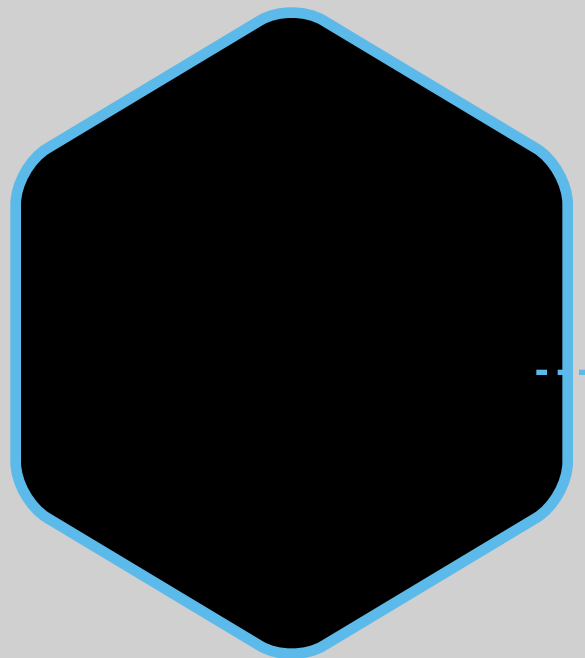
```
fn deploy --app projectfn
```

```
fn routes list projectfn
```

```
path image    endpoint  
/hello prpate1/hello:0.0.4  
localhost:8080/r/projectfn/hello
```



**Database (persistent storage)**

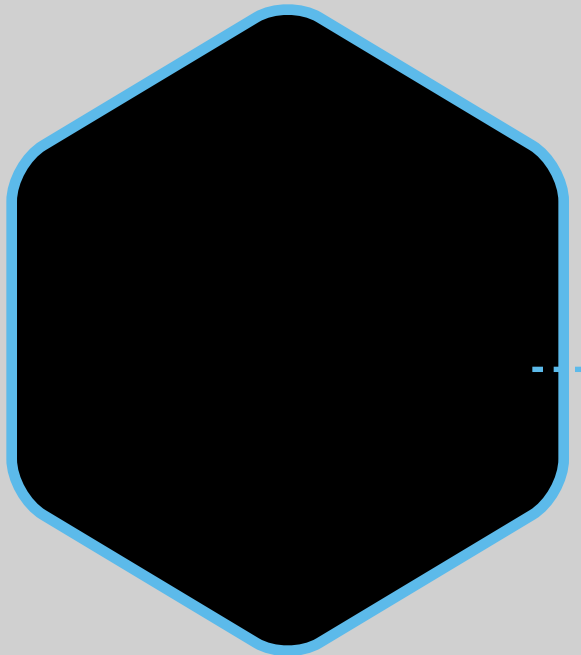


**Message Queues (msg'ing  
between function calls)**

**FN PROJECT**



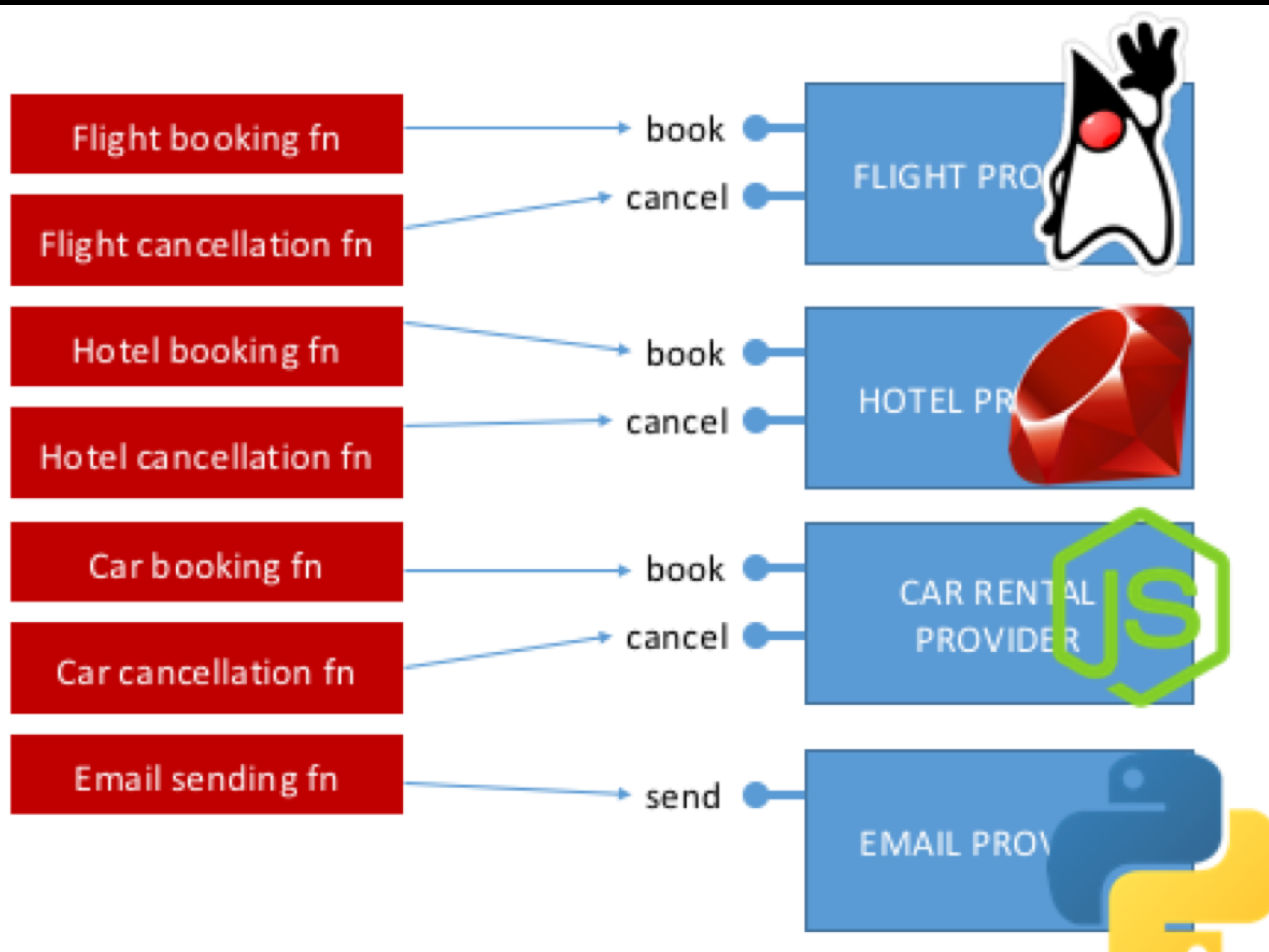
**Orchestrate multiple function calls (Fn Flow)**



**asynchronous functions  
(consumed by long-running components)**

**FN PROJECT**

# FN ORCHESTRATION



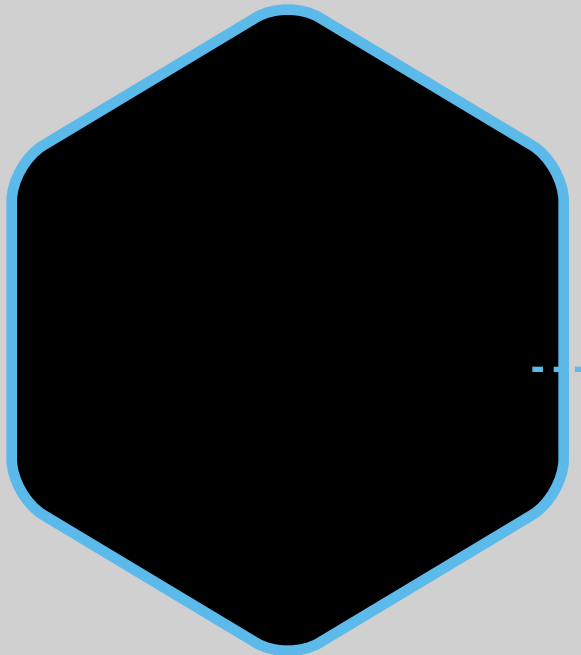


**DEMO**

**Open Whisk**



```
public static JsonObject  
main(JsonObject args);
```



**All OpenWhisk actions accept/  
return JSON objects**

# OPENWHISK



```
public class Hello {  
    public static JsonObject main(JsonObject args) {  
        String name = args.getAsJsonPrimitive("name").getString();  
        JsonObject response = new JsonObject();  
        response.addProperty("greeting", "Hello " + name + "!");  
        return response;  
    }  
}
```



```
gradle jar
```

```
wsk -i action create hello-java  
build/libs/hello-1.0.jar --main  
example.Hello
```

```
wsk -i action invoke /guest/hello-  
java --param name Pratik --blocking
```

# Comparison matrix

Framework	Langs supported	Deployment targets	Ease of Use
OpenWhisk			
Project Fn			
Open FaaS			
Kubeless			

Framework

Langs supported

Deployment  
targets

Ease of Use

OpenWhisk

Project Fn

Open FaaS

Kubeless

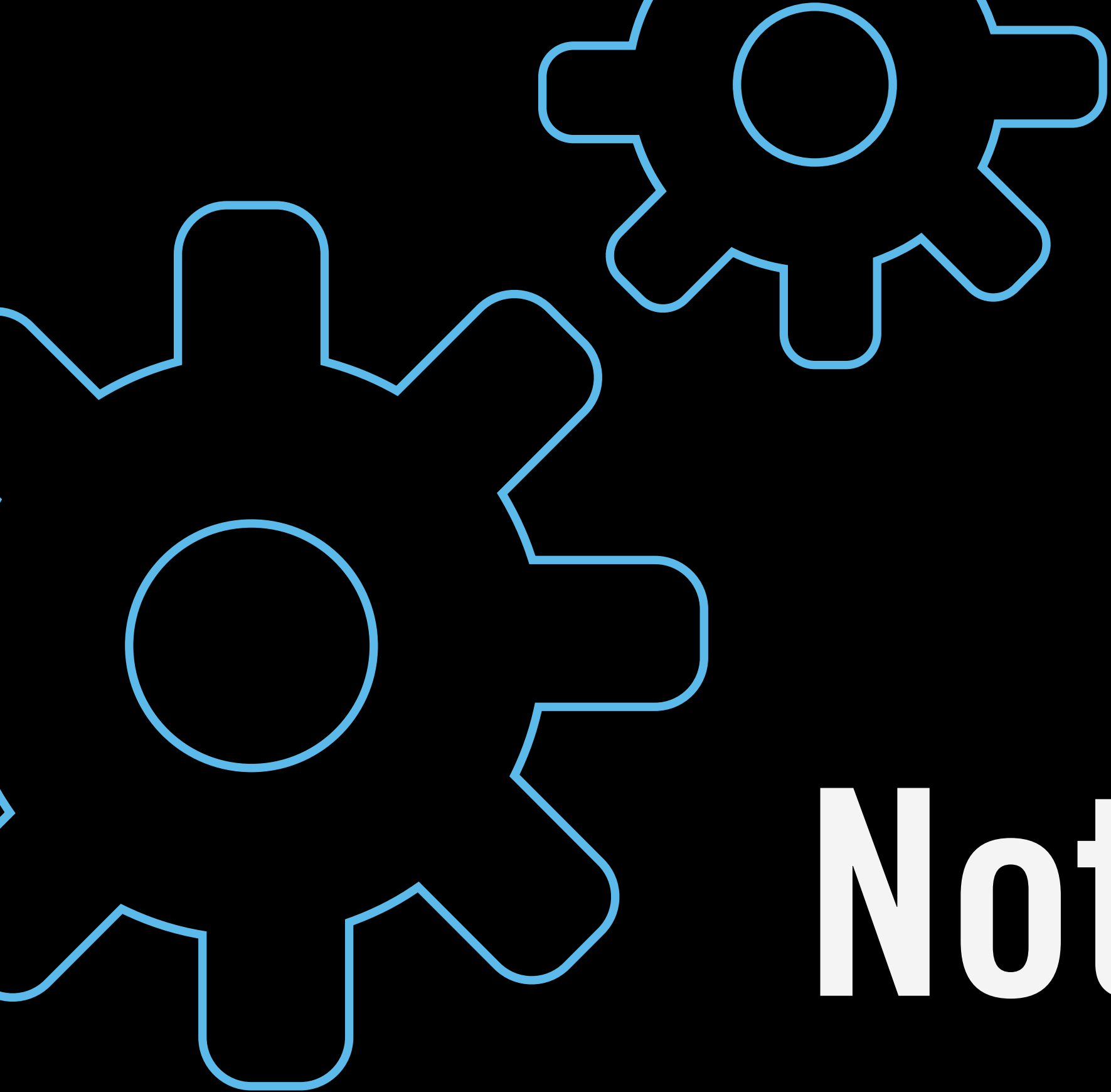
WHY THE HELL KNOWS?

Two large, light blue outline gears are positioned on the left side of the slide. The gear in the foreground is larger and partially cut off by the left edge. The gear behind it is smaller and fully visible.

**DESTROY OUR  
SERVERS?**



17:05:11:25\_CAM36



**Not quite yet!**

# WE (YOU?) ALREADY DOING THIS

my  
app



- *Twilio for SMS*
- *Stripe (payments)*
- *Image resize*
- *Send/format email*
- *Auth with Okta*
- *... lots more!*



**FAAS**

**As a developer, execution  
cost is my  
\*least important\* concern**

**@prpatel**



**FAAS**

**Time to market**



**FAAS**

**Cost of development**



**THANK YOU**

**follow me on twitter: @prpatel**

References:

<http://www.theregister.co.uk/2018/01/23/>

[serverless\\_exhilarating\\_terrifying\\_ridiculous\\_name/](http://www.theregister.co.uk/2018/01/23/serverless_exhilarating_terrifying_ridiculous_name/)

<https://www.theregister.co.uk/2017/12/12/serverless/>

<https://martinfowler.com/articles/serverless.html>

<https://medium.com/fnproject/serverless-sagas-with-fn-flow-d8199b608b12>

<https://medium.com/oracledevs/containers-vs-functions-51c879216b97>

**@prpatel**